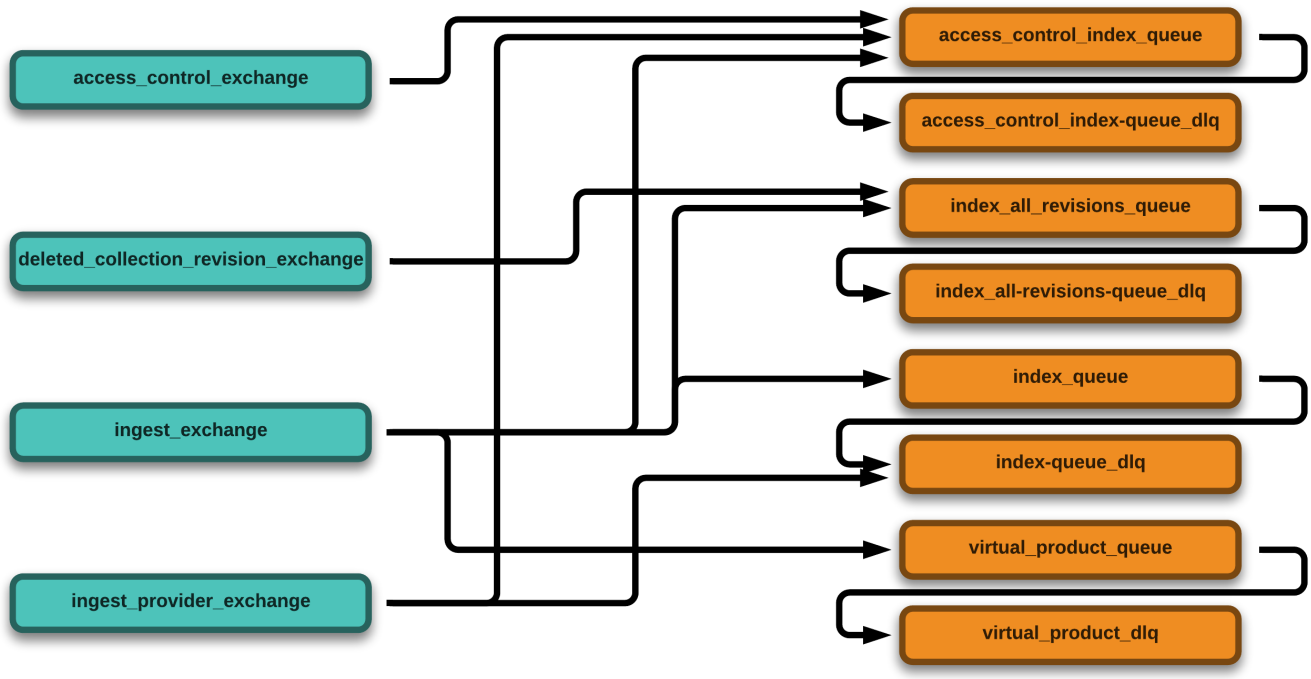
SNS/SQS Redesign

Existing Design

The existing design is given by the following diagram (dlq = dead letter queue)



Each primary (non-dlq) queue has a visibility timeout of 30, meaning that if a **listener** has not deleted a message from the queue within 30 seconds of receiving it then the queue will make it visible to other handlers. Each primary queue has a maximum read count of 5, meaning that after 5 reads by handlers without being deleted a message will be sent to the corresponding dead letter queue.

Primary queues are read by **listeners**, which are independent threads running in services like the indexer. **Listeners** are generic and rely on **han dlers**, which they call to process each message as shown in the following diagram:

Message types

The following is a list of all the message action types that we process.

TBD

Issues Encountered

1. Queues backed up during UAT cutover.
2. Queues backed up during Workload run test with manually running collection reindexing job.
3. Excessive access control backlog.

Root Causes

1. Provider collection reindexing takes much longer than the queue visibility timeout (30 seconds), which means each of these expensive and slow operations is retried by multiple threads until eventually one of them finishes. If it takes 10 minutes to complete then that message can end up being repeated by 20 threads.
2. Provider collection reindexing and single concept messages are worked together on the same queue, but collection reindexing takes significantly longer.
3. The access control queue receives many messages that it does not need to process because our exchanges are not set up such that access control can subscribe to only the events that it needs.